UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/675,181	09/30/2003	Michel Betancourt	RSW920030180US1 (120)	5912

46320 7590 08/23/2007

CAREY, RODRIGUEZ, GREENBERG & PAUL, LLP
STEVEN M. GREENBERG
950 PENINSULA CORPORATE CIRCLE
SUITE 3020
BOCA RATON, FL 33487

EXAMINER	
CAMPOS, YAIMA	

ART UNIT	PAPER NUMBER
2185	

MAIL DATE	DELIVERY MODE
08/23/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

80

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

MAILED

AUG 2 3 2007

Technology Center 2100

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 10/675,181
Filing Date: September 30, 2003
Appellant(s): BETANCOURT ET AL.

Scott D. Paul
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 12/26/06 appealing from the Office action mailed 7/24/06.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

5,933,840	Menon et al.	08-1999
2001/0023478	Ozawa et al.	09-2001
2002/0165870	Chakraborty et al.	11-2002

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 6, 8, 13 and 15 are rejected under 35 U.S.C. 102(b) as being anticipated by Menon et al. (US 5,933,840).

As per **claims 6 and 13**, Menon discloses "A method for detecting and remediating a memory leak," and "A machine readable storage having stored thereon a computer program for detecting and remediating a memory leak, the computer program comprising a routine set of instructions for causing the machine to perform the steps of:" as **["garbage collection of segments in a log-structured storage system" (Column 1, lines 10-11) and explains that "superseded data is referred to as garbage (or dead) and the corresponding disk area is referred to as a garbage block" (Column 3, lines 29-31). Menon also teaches that "the processor 102 includes one or more central processor units, such as a microprocessor, to execute programming instructions; random access memory (RAM) to contain application program instructions, system program instructions, and data; and an input/output controller to respond to read and write requests from executing applications" (Figure 1 and Column 6, lines 39-45)]** "the method comprising the steps of: establishing an aging value for an object instance created in memory;" **[With respect to this limitation, Menon discloses, "the age of a segment is determined with a time processor destage clock 132 that generates a timestamp value for a segment when that segment is filled in the memory segment buffer 128" (Column 10, lines 10-13)]**"resetting said aging value when said object instance is referenced by an executing process;" **[Menon discloses this concept as "age-queue buckets" wherein segments are grouped into "buckets where each bucket covers a range of utilization values" which "are organized as first in first out (FIFO) queues" (Column 12, lines 15-21). Menon also explains moving segments "to a different**

bucket as its utilization changes" and that whenever a segment "passes the age threshold, it is removed from the waiting list and enters the tail of the appropriate bucket determined by utilization" (Column 12, lines 34-45)] "incrementing said aging value during a garbage collection pass when said object instance had not been referenced by an executing process since a previous garbage collection pass;" **[With respect to this limitation, Menon discloses a process wherein "the age of a segment is defined as the difference between the current value of the destage clock and the timestamp of the segment itself. Therefore, GC-filled segment initially has an age equal to the age of the youngest segment that contributed tracks to it" (Column 10, lines 28-32); therefore, the age of a "GC-filled" segments would be incremented as a "clock is incremented" because the age of this segment would comprise a current time minus the timestamp value of the youngest segment in a GC-filled segment]**

"and, when said aging value exceeds a threshold value, processing said object instance as a loiterer" **[Menon discloses this limitation as "segments must wait in the DASD array for a minimum time equal to an age threshold before they can be considered for garbage collection" (Column 9, lines 60-62) and further teaches that "the array controller 112 selects target segments for garbage collection only after the segments pass the age threshold value, and selects segment in the order of smallest utilization rate" (Figures 1 and 2 and Column 11, lines 15-18)]**.

As per **claims 8 and 15**, Menon discloses "The method of claim 6," **[See rejection to claim 6 above]** "wherein said processing step comprises at least one of clearing at least one cache in memory," **[With respect to this limitation, Menon discloses "Non-volatile LRU managed cache 118" (Figure 1) and explains using a cache as "the LSA control unit 108 of the preferred embodiment includes both a non-volatile LSA data cache 118 and a memory segment buffer 128" (Column 8, lines 55-57)]** "and reporting said object instance as a loiterer in a log file" **[Menon discloses this concept as "age-queue buckets" wherein segments are grouped into "buckets where each bucket covers a range of utilization values" which "are organized as first in first out (FIFO) queues" (Column 12, lines 15-21). Menon also explains moving segments "to a different bucket as its utilization changes" and that whenever a segment "passes the age threshold, it is removed from the waiting list and enters the tail of the appropriate bucket determined by utilization" (Column 12, lines 34-45) as having queue logs of items selected for garbage collection]**.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. § 103 which forms the basis for all obviousness rejections set forth in this Office action:

> A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said

> subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.
>
> Subject matter developed by another person, which qualifies as prior art only under subsection (f) or (g) of section 102 of this title, shall not preclude patentability under this section where the subject matter and the claimed invention were, at the time the invention was made, owned by the same person or subject to an obligation of assignment to the same person.

Claims 1-4, 9-10, 12 and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Menon et al. (US 5,933,840) in view of Ozawa et al. (US 2001/0023478).

As per **claim 1**, Menon discloses "An autonomic memory leak detection and remediation system" as **["garbage collection of segments in a log-structured storage system" (Column 1, lines 10-11). Menon also explains that "superseded data is referred to as garbage (or dead) and the corresponding disk area is referred to as a garbage block" (Column 3, lines 29-31)]** "comprising: an autonomic garbage collector coupled to memory configured to store object instances which can be accessed by executing processes and which can be referenced by other object instances in said memory;" **[With respect to this limitation, Menon discloses, "processor 102 or host computer that communicates with an external information storage system 104 having N+1 direct access storage devices (DASD)" wherein "control unit 108 manages the transfer of data to and from the DASD array 106 so that periodically it considers segments for garbage collection" (Figure 1 and Column 6, lines 15-18 and 28-33)]**

"a tracing policy coupled to said autonomic garbage collector, said tracing policy specifying an aging threshold for a number of garbage collection passes during which an object instance in said memory is considered a loiterer when said object instance had not been accessed by one of said executing processes" **[Menon discloses this limitation as "segments must wait in the DASD array for a minimum time equal to an age threshold before they can be considered for garbage collection" (Column 9, lines 60-62) and further teaches that "the array controller 112 selects target segments for garbage collection only after the segments pass the age threshold value, and selects segment in the order of smallest utilization rate" (Figures 1 and 2 and Column 11, lines 15-18)].**

Menon does not disclose expressly does not disclose expressly having a "listing of exempt classes based upon which object instances are exempted from being labeled loiterers."

Ozawa discloses the concept of "listing of exempt classes based upon which object instances are exempted from being labeled loiterers" as **["cells are marked in three colors: white, black and off-white" and "white-marked cells can be judged to be *garbage* and black-marked cells cannot be judged to be *garbage*. In the sweep phase of PMGC, only white marked cells are collected" (Column 7, paragraph 00153)]**.

Menon et al. (US 5,933,840) in view of Ozawa et al. (US 2001/0023478) are analogous art because they are form the same field of endeavor of garbage collection.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to combine the garbage collection system which uses age of objects and usage information as taught by Menon and further exclude certain memory sections from being collected as garbage by marking them as "exempt" as taught by Ozawa.

The motivation for doing so would have been because Ozawa teaches that excluding certain memory sections from being collected as garbage by marking them as "exempt" **[solves "the problem that the live cells are collected by mistake" (Column 2, paragraph 0026) as "roots" are protected from being collected as garbage (Columns 1-2, paragraph 0019). Ozawa also teaches that "the number of pointers to be reported to the memory management unit can be reduced, and a process load can be reduced accordingly" (Column 3, paragraph 0040)]**.

Therefore, it would have been obvious to combine Ozawa et al. (US 2001/0023478) with Menon et al. (US 5,933,840) for the benefit of creating a garbage collection system/method to obtain the invention as specified in claim 1.

As per **claim 2**, the combination of Menon and Ozawa discloses "The system of claim 1," **[See rejection to claim 1 above]** "wherein said memory managed through a virtual machine" **[With respect to this limitation, Menon teaches "In this way, one or more logical (virtual) devices are mapped onto the actual DASDs of the array 106 by the array control unit 108" (Column 9, lines 41-45)]** wherein "said memory is a heap" **[With respect to this limitation, Ozawa discloses "a sweep phase, the entire heap is scanned and unmarked cells are collected" wherein "a mark phase and a**

sweep phase are alternately repeated" in a "mark and sweep" garbage collection method (Column 1, paragraph 0017) and explains that "in the description of garbage collection, a memory resource to be managed is called a heap" wherein "the heap is composed of units of data strings called cells (of objects). Each cell can include pointers pointing to another cell and another data" and explains that "the number of pointers to be reported to the memory management unit can be reduced, and a process load can be reduced accordingly" (Column 3, paragraph 0040) when using a heap].

As per <u>**claim 3**</u>, the combination of Menon and Ozawa discloses "the system of claim 1" **[See rejection to claim 1 above]**

"wherein said autonomic garbage collector comprises a mark and sweep garbage collector" **[Menon discloses this concept as a garbage collection method for "marking" objects for garbage collection and later "sweeping" or collecting these objects as the system "considers a segment for garbage collection only after the segment has been located in DASD for the selected age threshold value" (Column 5, lines 28-30) as marking segments for garbage collection. Menon also disclose that "after the segments pass the age threshold value," then "array controller 112" selects "segments in the order of smallest utilization rate" (Column 11, lines 15-18) as explaining that segments are later collected/swept]**

"modified both to manage aging values associated with object instances in said memory"

[With respect to this limitation, Menon discloses "the age of a segment is determined with a time processor destage clock 132 that generates a timestamp value for a segment when that segment is filled in the memory segment buffer 128" (Column 10, lines 10-13)]

"and also to compare said aging values to said aging threshold to identify loiterers" **[Menon discloses this limitation as "segments must wait in the DASD array for a minimum time equal to an age threshold before they can be considered for garbage collection" (Column 9, lines 60-62) and further teaches that "the array controller 112 selects target segments for garbage collection only after the segments pass the age threshold value, and selects segment in the order of smallest utilization rate" (Figures 1 and 2 and Column 11, lines 15-18). Additionally, in order to perform garbage collection, there must be some sort of marking or identification of sections of memory selected for garbage collection in order to later sweep or collect garbage]**.

As per **claim 4**, the combination of Menon and Ozawa discloses "The system of claim 1," **[See rejection to claim 1 above]** "wherein said tracing policy further comprises a specification for at least one action to be undertaken upon detecting a loiterer" **[With respect to this limitation, Menon discloses "the array controller 112 selects target segments for garbage collection only after the segments pass the age threshold value, and selects segment in the order of smallest utilization rate"**

(Figures 1 and 2 and Column 11, lines 15-18) as "garbage collecting" segments that are identified as "loiterers"].

As per **claims 9 and 16**, Menon discloses the system of claims 6 and 13, **[See rejection to claims 6 and 13 above]** but does not disclose expressly "listing of exempt classes based upon which object instances are exempted from being labeled loiterers."

Ozawa discloses the concept of "listing of exempt classes based upon which object instances are exempted from being labeled loiterers" as **["cells are marked in three colors: white, black and off-white" and "white-marked cells can be judged to be *garbage* and black-marked cells cannot be judged to be *garbage*. In the sweep phase of PMGC, only white marked cells are collected" (Column 7, paragraph 00153)]**.

Menon et al. (US 5,933,840) in view of Ozawa et al. (US 2001/0023478) are analogous art because they are form the same field of endeavor of garbage collection.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to combine the garbage collection system which uses age of objects and usage information as taught by Menon and further exclude certain memory sections from being collected as garbage by marking them as "exempt" as taught by Ozawa.

The motivation for doing so would have been because Ozawa teaches that excluding certain memory sections from being collected as garbage by marking them as "exempt" **[solves "the problem that the live cells are collected by mistake" (Column 2, paragraph 0026) as "roots" are protected from being collected as**

garbage (Columns 1-2, paragraph 0019). Ozawa also teaches that "the number of pointers to be reported to the memory management unit can be reduced, and a process load can be reduced accordingly" (Column 3, paragraph 0040)].

Therefore, it would have been obvious to combine Ozawa et al. (US 2001/0023478) with Menon et al. (US 5,933,840) for the benefit of creating a garbage collection system/method to obtain the invention as specified in claims 9 and 16.

As per **claim 10**, Menon discloses "An autonomic memory leak detection and remediation method" as **["garbage collection of segments in a log-structured storage system" (Column 1, lines 10-11); explains that "superseded data is referred to as garbage (or dead) and the corresponding disk area is referred to as a garbage block" (Column 3, lines 29-31)]**. Menon also teaches modifying a mark and sweep garbage collection as **[a garbage collection method which for "marking" objects for garbage collection and later "sweeping" or collecting these objects as the system "considers a segment for garbage collection only after the segment has been located in DASD for the selected age threshold value" (Column 5, lines 28-30) as marking segments for garbage collection. Menon also disclose that "after the segments pass the age threshold value," then "array controller 112" selects "segments in the order of smallest utilization rate" (Column 11, lines 15-18) as explaining that segments are later collected/swept]** "managing aging values associated with object instances created in memory" as **["the age of a segment is determined with a time processor destage clock 132 that generates a timestamp**

value for a segment when that segment is filled in the memory segment buffer 128" (Column 10, lines 10-13)] "and, processing as loiterers selected ones of said object instances having aging values which exceed a predetermined threshold" **[Menon discloses this limitation as "segments must wait in the DASD array for a minimum time equal to an age threshold before they can be considered for garbage collection" (Column 9, lines 60-62) and further teaches that "the array controller 112 selects target segments for garbage collection only after the segments pass the age threshold value, and selects segment in the order of smallest utilization rate" (Figures 1 and 2 and Column 11, lines 15-18). Additionally, in order to perform garbage collection, there must be some sort of marking or identification of sections of memory selected for garbage collection in order to later sweep or collect garbage]**

"where said selected ones of said object instances have aging values which exceed a predetermined threshold" **[Menon discloses this limitation as "segments must wait in the DASD array for a minimum time equal to an age threshold before they can be considered for garbage collection" (Column 9, lines 60-62) and further teaches that "the array controller 112 selects target segments for garbage collection only after the segments pass the age threshold value, and selects segment in the order of smallest utilization rate" (Figures 1 and 2 and Column 11, lines 15-18)]**..

Menon does not disclose expressly does not disclose expressly having "said processing step comprise the step of processing as loiterers selected ones of said object instances not belonging to an exempt class."

Ozawa discloses the concept of "said processing step comprise the step of processing as loiterers selected ones of said object instances not belonging to an exempt class" as **["cells are marked in three colors: white, black and off-white" and "white-marked cells can be judged to be *garbage* and black-marked cells cannot be judged to be *garbage*. In the sweep phase of PMGC, only white marked cells are collected" (Column 7, paragraph 00153)]**.

Menon et al. (US 5,933,840) in view of Ozawa et al. (US 2001/0023478) are analogous art because they are form the same field of endeavor of garbage collection.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to combine the garbage collection system which uses age of objects and usage information as taught by Menon and further exclude certain memory sections from being collected as garbage by marking them as "exempt" as taught by Ozawa.

The motivation for doing so would have been because Ozawa teaches that excluding certain memory sections from being collected as garbage by marking them as "exempt" **[solves "the problem that the live cells are collected by mistake" (Column 2, paragraph 0026) as "roots" are protected from being collected as garbage (Columns 1-2, paragraph 0019). Ozawa also teaches that "the number of pointers to be reported to the memory management unit can be reduced, and a process load can be reduced accordingly" (Column 3, paragraph 0040)]**.

Therefore, it would have been obvious to combine Ozawa et al. (US 2001/0023478) with Menon et al. (US 5,933,840) for the benefit of creating a garbage collection system/method to obtain the invention as specified in claim 10.

As per **claim 12**, the combination of Menon and Ozawa discloses "The method of claim 10 **[See rejection to claim 10 above]** "wherein said processing step comprises at least one of clearing at least one cache in memory," **[With respect to this limitation, Menon discloses "Non-volatile LRU managed cache 118" (Figure 1) and explains using a cache as "the LSA control unit 108 of the preferred embodiment includes both a non-volatile LSA data cache 118 and a memory segment buffer 128" (Column 8, lines 55-57)]** "and reporting said object instance as a loiterer in a log file" **[Menon discloses this concept as "age-queue buckets" wherein segments are grouped into "buckets where each bucket covers a range of utilization values" which "are organized as first in first out (FIFO) queues" (Column 12, lines 15-21). Menon also explains moving segments "to a different bucket as its utilization changes" and that whenever a segment "passes the age threshold, it is removed from the waiting list and enters the tail of the appropriate bucket determined by utilization" (Column 12, lines 34-45) as having queue logs of items selected for garbage collection]**.

Claims 7 and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Menon et al. (US 5,933,840) in view of Chakraborty et al. (US 2002/0165870).

As per **claims 7 and 14**, Menon discloses the system/method of claims 6 and 13, **[See rejection to claims 6 and 13 above]** but fails to disclose expressly "locating equivalent object instances in said memory; and, processing said equivalent object instances in said memory as loiterers."

Chakraborty discloses "locating equivalent object instances in said memory; and, processing said equivalent object instances in said memory as loiterers" as **["the most recently used time stamp for the repeated nodes becomes the time stamp for all of those nodes;" then "garbage collector creates an intermediate data structure" which "holds one entry for each repeated node. The least recently used of all entries in the intermediate data structure is chosen and then, all of those repeated entries in the node table are removed by the garbage collector" (Column 3, paragraph 0050, lines 3-12)]**.

Menon et al. (US 5,933,840) and Chakraborty et al. (US 2002/0165870) are analogous art because they are form the same field of endeavor of garbage collection.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to combine the garbage collection system which uses age of objects and usage information as taught by Menon and further process equivalent or repeated object instances as "loiterers" or garbage as taught by Chakraborty.

The motivation for doing so would have been because Chakraborty discloses that removing or garbage collecting repeated object instances allows the system to **["reduce latency" and "reduce traffic" as multiple instances of the same object are removed from cache; therefore, since these object instances are "retrieved from a server once, it reduces the amount of bandwidth used by a client" (Column 2, paragraphs 0016-0018)]**.

Therefore, it would have been obvious to combine Chakraborty et al. (US 2002/0165870) with Menon et al. (US 5,933,840) for the benefit of creating a garbage collection system/method to obtain the invention as specified in claims 7 and 14.

(10) Response to Argument

The Rejection of Claims 6, 8, 13, and 15 Under 35 U.S.C. 102 for Anticipation Based Upon Menon

Claims 6 and 13:

Appellant argues that Menon does not disclose "resetting said aging value when said object instance is referenced by an executing process" as the Examiner is inappropriately construing the age-queue buckets disclosed by Menon as being "resetting said aging value" and that instead, Menon employs utilization rates, which are not comparable to the claimed aging values.

These arguments were fully considered but were not persuasive.

Menon discloses "resetting said aging value when said object instance is referenced by an executing process" as [**"garbage collection of segments in a log-structured storage system" (Column 1, lines 10-11) and explains that "superseded data is referred to as garbage (or dead) and the corresponding disk area is referred to as a garbage block" (Column 3, lines 29-31) "the array controller 112 selects target segments for garbage collection only after the segments pass the age threshold value, and selects segment in the order of smallest utilization rate" (Figures 1 and 2 and Column 11, lines 15-18) wherein, "age-queue buckets" are used to group segments into "buckets where each bucket covers a range of utilization values" which "are organized as first in first**

out (FIFO) queues" (Column 12, lines 15-21). Menon also explains moving segments "to a different bucket as its utilization changes" and that whenever a segment "passes the age threshold, it is removed from the waiting list and enters the tail of the appropriate bucket determined by utilization" (Column 12, lines 34-45) wherein "if a candidate segment is to be selected for garbage collection, then the segment at the head of the lowest-numbered (lowest utilization range bucket) non-empty bucket is used first" (Col. 12, lines 51-67)]. Appellant should note that utilization of a memory segment occurs when executing processes reference/use a memory segment. In response to changes in utilization, segments are moved to different "age-queue buckets" wherein each "age-queue bucket" comprises a different age/aging value since the memory segment has been used/referenced by an executing process. Menon selects segments for garbage collection from the least-used (lowest utilization range or lowest numbered) "age queue bucket." When a segment from the lowest numbered age-queue bucket is referenced by an executing process, it's utilization changes and this segment is moved from this lowest numbered age-queue bucket (*which represents a bucket having the oldest/highest age value since a process has referenced/used its contents*) to a higher numbered bucket (*which represents a bucket having a smaller age value since a process has referenced/used its contents*). Therefore, as a memory segment is moved from one age-queue bucket to another in response to changes in utilization, the utilization value or age value since that segment has been referenced by an executing process is changed; therefore, it is reset. When a segment is selected for garbage collection, it is selected from the lower numbered age-

queue bucket (*which represents a bucket having the oldest/highest age value since a process has referenced/used its contents*). Menon, in essence, resets the age of a segment for garbage collection, achieving the functionality of "resetting said aging value when said object instance is referenced by an executing process."

The Examiner would also like to refer Appellant to **["one bucket 402a will be designated for segments having utilization rates from zero to 0.1, the next bucket 402b will be designated for buckets with utilization rates greater than 0.1 and less or equal to 0.2, the next bucket will be for rates greater than 0.2 and less than 0.3, and so forth, to a bucket 402c for rates u where $0.9<u\leq 1$" (Col. 12, lines 11-33; Figure 4) where each age-queue bucket is provided with identifiers which are interpreted as the claimed *aging values.* Menon explains "age-queue buckets" are used to group segments into "buckets where each bucket covers a range of utilization values" which "are organized as first in first out (FIFO) queues" (Column 12, lines 15-21). Menon also explains moving segments "to a different bucket as its utilization changes" and that whenever a segment "passes the age threshold, it is removed from the waiting list and enters the tail of the appropriate bucket determined by utilization" (Column 12, lines 34-45) wherein "if a candidate segment is to be selected for garbage collection, then the segment at the head of the lowest-numbered (lowest utilization range bucket) non-empty bucket is used first" (Col. 12, lines 51-67)]**. Utilization of a memory segment occurs when executing processes reference/use a memory segment. In response to changes in utilization, segments are moved to different "age-queue buckets" (having identifiers which

represent "aging values") since each "age-queue bucket" comprises a different age/aging value since the memory segment has been used/referenced by an executing process. As the utilization of a memory segment changes, this memory segment is moved to an age-queue bucket having a different aging value. As the value of the age-queue bucket where the segment resides changes, the aging value since an executing process has referenced that memory segment is reset; therefore Menon discloses "resetting said aging value when said object instance is referenced by an executing process" as claimed.

Furthermore, the Examiner would like to respectfully submit that the claims do not contain any requirement/limitation dictating how the resetting process occurs or adding any details to the claimed "resetting."

Appellant argues that Menon does not disclose "incrementing said aging value during a garbage collection pass when said object instance had not been referenced by an executing process since a previous garbage collection pass," as Menon teaches that an age of a segment can be determined but does not disclose the above-identified limitation.

In response to Appellant's arguments, the arguments were fully considered but were not deemed persuasive.

First, the Examiner would like to point out that Menon discloses this limitation as **["the age of a segment is defined as the difference between the current value of the destage clock and the timestamp of the segment itself. Therefore, GC-filled**

segment initially has an age equal to the age of the youngest segment that contributed tracks to it" (Column 10, lines 28-32); therefore, the age of a "GC-filled" segments would be incremented as a "clock is incremented" because the age of this segment would comprise a current time minus the timestamp value of the youngest segment in a GC-filled segment and explains "the present invention manages an information storage system of a computer to provide a system in which target segments are selected for garbage collection only if their age in the information storage system exceeds an age threshold value" (Col. 5, lines 19-24) and explains that "segments recently used filled by write operations would wait an age threshold amount of time before they are allowed to become candidates for garbage collection" (Col. 9, lines 65 – Col. 10, line 3)]. Therefore, for every garbage collection pass when a segment is not filled by write operations, the segment's age must be incremented in order for the segment to be able to wait to reach an age threshold since this segment has not been filled by write operations and be considered for garbage collection. Menon discloses that segments are not considered for garbage collection until they reach an age threshold since they have been filled by write operations; therefore, Menon discloses "incrementing said aging value during a garbage collection pass when said object instance had not been referenced by an executing process since a previous garbage collection pass" as required by the claims.

Appellant argues the following:

"On page 6 of the Office Action, the Examiner asserted that "Menon discloses an equivalent process." Whether or not Menon discloses an equivalent process is immaterial to a rejection under 35 U.S.C. § 102, which requires the identical disclosure of each element. A finding of equivalency does equate to a finding

of anticipation. Moreover, a finding of equivalency is only useful in obviousness rejections. In this regard, the Examiner is also referred to M.P.E.P. § 2144.06 and the paragraph entitled "ART RECOGNIZED EQUIVALENCE FOR THE SAME PURPOSE." Since the Examiner has failed to establish that Menon identically discloses the above-identified limitation, Menon further fails to identically disclose the claimed invention, as recited in claims 6 and 13, within the meaning of 35 U.S.C. § 102.

Although the Examiner addressed other arguments made by Appellants on pages 15-17 of the Second Office Action, the Examiner did not address the above argument. Instead, the Examiner rephrased the statement of the rejection to remove the term "equivalent" from the phrase "Menon discloses an equivalent process.""

This argument has been fully considered by the Examiner, but is not persuasive.

The Examiner would like to respectfully point out that in the Non-Final rejection mailed on 2/10/06; the Examiner did not intend to raise a 103 Obviousness type rejection when the word "equivalent" was used; accordingly, this word was deleted from the 102 Anticipation type rejection in the Final rejection mailed on 7/24/06. The Examiner submits that Menon reads on the claimed process as explained above.

Appellant has not addressed any new points regarding the rejection of dependent claims 8 and 15. Therefore, examiner submits that if the rejection of the independent claims 6 and 13 is deemed proper, the rejection of claims 8 and 15 should be upheld.

The Rejection of Claims 1-4, 9-10, 12, and 16 Under 35 U.S.C. 103 for Obviousness Based Upon Menon in view of Ozawa

Claim 1

Appellant argues that the combination of Menon and Ozawa does not disclose "said tracing policy further comprises a listing of exempt classes based upon which object instances are exempted from being labeled as loiterers" as the use of colors to

mark the cells, as taught by Ozawa is not comparable to the claimed exempt classes as "the color fields are an indication of a particular status of a cell, not whether that cell is within a particular class" (Notice of Appeal, Page 8) and that "even if the teaching by Ozawa of cells marked by black could be considered a class, Ozawa only teaches the single class. In contrast, the claimed invention recites a listing of exempt classes" (i.e., a plurality of classes)" (Notice of Appeal, Page 9).

These arguments have been fully considered by the Examiner but are not persuasive.

First, the Examiner would like to point out that Ozawa discloses this limitation as **["In the description of GC, a memory resource to be managed is called a "heap". The heap is composed of units of data strings called cells (or objects)" (Page 1, Paragraph 0005) and explains that "This cell also has a color field... The color field displays the color of a mark attached to the cell. In this preferred embodiment, as a rule, three colors of white, black and off-white are used for the color attached to a color field" (Page 7, Paragraph 0143) and refers to color marking as "classification" (Page 14, Paragraph 0288) wherein "white-marked cells can be judged to be *garbage* and black-marked cells cannot be judged to be *garbage*. In the sweep phase of PMGC, only white marked cells are collected" (Column 7, paragraph 00153)]**. A class is a set, collection, group, or configuration containing members regarded as having certain attributes in common; a kind or category. Applicant should note that all memory cells marked in black in Ozawa comprise a listing/class/configuration of elements having an attribute in common, that

all memory cells marked in off-white in Ozawa also comprise a listing/class/configuration of elements having an attribute in common and in the same manner, all the white cells comprise a listing/class/configuration of elements having an attribute in common. In this case, the attribute in common comprises the color classifications of these cells/objects. Accordingly, since all cells/objects classified with the black or off-white colors will not be garbage collected during a sweep phase, these classes of cells are exempt being collected during a garbage collection cycle. Therefore, since only white cells are garbage collected, Ozawa discloses at least two classes of cells that are exempt from garbage collection; thereby disclosing a garbage collection tracing policy comprising "a listing of exempt classes based upon which object instances are exempted from being labeled loiterer."

Additionally, Appellant should note that Ozawa discloses [**"In the description of GC, a memory resource to be managed is called a "heap". The heap is composed of units of data strings called cells (or objects)" (Page 1, Paragraph 0005) and explains that "This cell also has a color field... The color field displays the color of a mark attached to the cell. In this preferred embodiment, as a rule, three colors of white, black and off-white are used for the color attached to a color field" (Page 7, Paragraph 0143) and refers to color marking as "classification" (Page 14, Paragraph 0288) wherein "white-marked cells can be judged to be *garbage* and black-marked cells cannot be judged to be *garbage*. In the sweep phase of PMGC, only white marked cells are collected" (Column 7, paragraph 00153)**]. According to this disclosure, each memory cell can interpreted to belong to a

separate class wherein black-marked cells cannot be judged to be garbage and are exempt from garbage collection. Each cell belongs to a separate class and all cells marked in black comprise a listing of classes that are exempt from being subject to garbage collection. Therefore, Ozawa discloses a garbage collection tracing policy comprising "a listing of exempt classes based upon which object instances are exempted from being labeled loiterer."

Appellant's arguments directed towards the rejection of claim 10 reiterate the deficiencies Appellant made in the rejection of the independent claim 1 and do not address any new points. Therefore, examiner submits that if the rejection of the independent claim 1 is deemed proper, the rejection of claims 10 should be upheld.

Appellant has not addressed any new points regarding the rejection of dependent claims 3-4, 9, 12 and 16. Therefore, examiner submits that if the rejection of the independent claim 1 is deemed proper, the rejection of claims 3-4, 9, 12 and 16 should be upheld.

<u>The Rejection of Claims 7 and 14 Under 35 U.S.C. 103 for Obviousness Based Upon Menon in view of Ozawa and Chakraborty</u>

Appellant's arguments directed towards the rejections of claims 7 and 14 reiterate the deficiencies Appellant made in the rejection of the independent claim 6

above. Therefore, examiner submits that if the rejection of the independent claim 6 is deemed proper, the rejection of claims 7 and 14 should be upheld.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Yaima Campos

Examiner

Conferees:

Sanjiv Shah

Supervisory Patent Examiner

SANJIV SHAH
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Lynne Browne

~~Supervisory Patent Examiner~~
APPEAL PRACTICE SPECIALIST, TQAS
TECHNOLOGY CENTER 2100